UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

(2)

1a. REPORT SECURITY CLASSIFICATION Ur	1b. RESTRICTIVE MARKINGS
2a.	3. DISTRIBUTION/AVAILABILITY OF REPORT: Approved for public release; distribution unlimited.
2b.	
4. P	5. MONITORING ORGANIZATION REPORT NUMBER(S): #66
6a. NAME OF PERFORMING ORGANIZATION: Institute for Brain and Neural Systems — 6b. OFFICE SYMBOL (If applicable)	7a. NAME OF MONITORING ORGANIZATION: Personnel and Training Research Programs, Office of Naval Research (Code 1142PT)
6c. ADDRESS (City, State, and ZIP Code): Brown University, Providence, Rhode Island 02912	7b. ADDRESS (City, State and ZIP Code): 800 North Quincy Street, Arlington, VA 22217-5000
8a. NAME OF FUNDING/SPONSORING ORGANIZATION — 8b. OFFICE SYMBOL (If applicable)	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER: N00014-91-J-1316
8c. ADDRESS (City, State, and ZIP Code)	10. SOURCE OF FUNDING NUMBERS: PROGRAM ELEMENT NO / PROJECT NO / TASK NO / WORK UNIT ACCESSION NO

AD-A264 936

C E D 93 A

11. TITLE (Include Security Classification)
A Soft-Competitive Splitting Rule for Adaptive Tree-Structured Neural Networks.

12. PERSONAL AUTHOR(S)
Michael P. Perrone

13a. TYPE OF REPORT	13b. TIME COVERED	14. DATE OF REPORT (Year, Month, Day)	15. PAGE COUNT
Technical Report	FROM ____ TO ____	May 17, 1993	Five

16. SUPPLEMENTARY NOTATION

17. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)
FIELD	GROUP	SUB-GROUP	Soft-Competition, CART, Neural Networks
05	08		

19. ABSTRACT (Continue on reverse if necessary and identify by block number)

An algorithm for generating tree structured neural networks using a soft-competitive recursive partitioning rule is described. It is demonstrated that the algorithm grows robust, honest estimators. Preliminary results on a 10 class, 240 dimensional OCR classification task are presented which show that the tree out-performs backpropagation. Arguments are made which suggest why this should be the case. The connection of the soft-competitive splitting rule to the twoing rule is described.

93 5 039

93-11615

20. DISTRIBUTION/AVAILABILITY OF ABSTRACT	21. ABSTRACT SECURITY CLASSIFICATION
☒ UNCLASSIFIED/UNLIMITED ☐ SAME AS RPT ☐ DTIC USERS	Unclassified
22a. NAME OF RESPONSIBLE INDIVIDUAL: Dr. Joel Davis	22b. TELEPHONE (Include Area Code): (703) 696-4744 — 22c. OFFICE SYMBOL

DD FORM 1473, 84 MAR — 83 APR edition may be used until exhausted. All other editions are obsolete.

SECURITY CLASSIFICATION OF THIS PAGE

A Soft-Competitive Splitting Rule for Adaptive Tree-Structured Neural Networks *

Michael P. Perrone
Physics Department and Center for Neural Science
Box 1843, Brown University
Providence, RI 02912
Email: mpp@cns.brown.edu

June 6, 1992

Abstract

An algorithm for generating tree structured neural networks using a soft-competitive recursive partitioning rule is described. It is demonstrated that this algorithm grows robust, honest estimators. Preliminary results on a 10 class, 240 dimensional OCR classification task are presented which show that the tree out-performs backpropagation. Arguments are made which suggest why this should be the case. The connection of the soft-competitive splitting rule to the twoing rule is described.

1 Introduction

In even the simplest cases, gradient descent algorithms such as backpropagation are prone to sub-optimal behavior due to spurious local minima (Sontag and Sussmann, 1988). This problem is related to the strong interference effects that a single backprop net will experience when it is trained to perform many different sub-tasks (Jacobs et al., 1991). This interference gives rise to spurious local minima which impair the net's learning and generalization. It is therefore desirable to sub-divide complex tasks into sub-tasks which are simpler, since the networks needed to process these sub-tasks will necessarily be less complex than a general purpose net and will therefore be faced with fewer local minima.

This notion of sub-division was first implemented successfully in the CART algorithm (Breiman et al., 1984). This method implemented very simple splits but was succesfully due to the introduction of pruning which prevented the solutions from becoming biased to the training data. If one knows of an *a priori* sub-task breakdown, sub-nets can be constructed by hand (Hampshire and Waibel, 1989). Otherwise, the sub-tasks must be extracted from the data. Identifying sub-tasks often requires the extraction of new features which has been done in an unsupervised manner (Intrator, 1991). Also sub-tasks can be identified as part of the learning process (Reilly et al., 1987; Jacobs et al., 1990; Jacobs et al., 1991; Sanger, 1991). It is also possible to construct a CART-like tree using perceptrons to perform the partitioning (Sankar and Mammone, 1991).

*Research was supported by the National Science Foundation, the Army Research Office, and the Office of Naval Research.

In this paper we present an algorithm which, unlike the algorithms discussed above, untilizes a modified version of the original CART twoing rule for splitting (Perrone, 1991). This new splitting rule is a soft-competitive (Hinton and Nowlan, 1990) neural net anolog of the twoing rule which constructs splits based on a minimization of the interference between sub-tasks.

We use the CART methodology of top-down/bottom-up pruning when growing our network to assure that we grow right-sized trees which are honest estimators for the underlieing probability distributions. This helps prevent the tree architecture from becoming biased to the training data. The additional problem of over-fitting at the sub-network level is avoided by using cross-validation as a stopping criterion for the training of the tree node networks.

The soft-competition splitting rule is used to decide how to divide tasks into sub-tasks. The soft-competition in the splitting rule allows us to easily determine which sub-tasks are most distinct and therefore is trying to maximize the reduction in interference with each successive split.

In section 2, the soft-competition splitting rule is explained. In section 3, examples are given which demonstrate the algorithm's ability to avoid local minima by smart choice of sub-tasks; and preliminary results on a real-world character recognition task are presented. Section 4 contains a summary and conclusions.

2 The Soft-Competition Splitting Rule

Let p_l^i be the lth pattern from class i at a given tree node. Let $f_j(p_l^i)$ be the value of the jth sigmoidal output unit of a backprop net for the lth pattern from the ith class. Define the *confusion matrix*, m_{ij}, as follows:

$$m_{ij} \equiv \frac{\sum_l f_j(p_l^i)}{\sum_{j,l} f_j(p_l^i)}.$$

The confusion matrix is thus measuring the overall signal of the ith class in the jth output of the backprop net. Note that if the net were a perfect classifier, then the confusion matrix would be the identity matrix. Thus, the off-diagonal terms can be thought of as a measure of the confusion between classes.

A *class partition* is defined as a grouping of the labels of the classes present at a given tree node into two, distinct subgroups α and β of labels. For a given class partition, we can define a *confusion measure*, $M(\alpha, \beta)$ as the amount of confusion existing between the two groups in the class partition, thus:

$$M(\alpha, \beta) \equiv \frac{1}{N_\alpha N_\beta} \sum_{i\in\alpha, j\in\beta} m_{ij},$$

where N_α and N_β are the number of classes in α and β, respectively; m_{ij} is an element from the confusion matrix; and α and β are the subgroups of the partition.

Now, the α and β can be thought of as sub-tasks, so the confusion measure is a measure of the interference between the two sub-tasks. This gives us a very convenient way of choosing sub-tasks: we simply minimize the confusion measure over all of the partitions. This can be done with an exhaustive search, or, if the number of classes is too large, one can use a simulated annealing algorithm on the confusion measure (Press et al., 1987).

In practice, one would train a backprop net on the full problem at a particular tree node, use this net to generate a confusion matrix, use the confusion matrix to generate a class partition and then train a new net to perform the partitioning to the children nodes of the tree.

The importance of the soft-competition that is inherent in the definition of the confusion matrix is the following. Consider the situation in which the outputs for two classes are very similar but the correct class is usually greater than the other, and consider a third class output which is usually much less than both. If we use hard competition by setting all but one of the outputs to one, we throw away information that tells us that the relationship between the three classes is not the same.

It is also interesting to note that the splitting rule above can be thought of a a neural net implementation of the twoing rule described by Breiman (Breiman et al., 1984).

3 Partitioning Examples

In this section, we present two toy classification tasks and one real-world classification task. We discuss why backprop has difficulty and we show how the tree-structured algorithm avoids these problems. It should be noted that we are not claiming that backprop can not solve these problems, but rather that the tree algorithms solves them more easily. This is a characteristic that the tree algorithm maintains in high dimensional problems where spurious minima really start to effect the performance of backprop.

Figure 1: Two, two-dimensional deterministic classification problems. Classes are labelled 1, 2 and 3. Sample points are uniformly distributed within class regions.

The classification tasks we will consider are depicted in figure 3. Note that in principle the minimal size net needed to solve the first task has a single hidden layer of four hidden units, while the second task requires two hidden layers of two hidden units each. In practice, however, the gradient descent of backpropagation will frequently leave the networks in the sub-optimal local minima depicted by figure 3 since the local minima are broad and the global minima are narrow.

The tree network constructed by the algorithm in this paper had no difficulty finding the global minimum everytime. (See figure 3.) ssince the splitting nets used by the tree algorithm were always less complex than the single backprop nets. In the first case, a minimum backprop solution requires a hidden layer with four hidden units, while the tree solved the problem with perceptrons. In the second case, the minimum backprop architecture requires two hidden layers of two hidden units each, while the tree solved the problem with two hidden units in a single hidden layer. Thus the tree was less prone to spurious local minima.

The tree algorithm was also tested on a large real-world pattern classification problem. The numbers '0' through '9' were taken from the NIST OCR database and were used as a classification

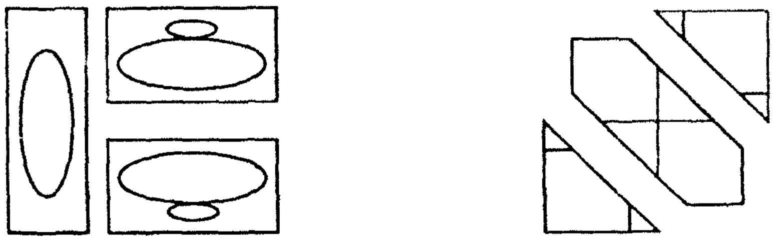

Figure 2: Local minima solutions which backprop finds. Note that in each case backprop is not performing optimally.

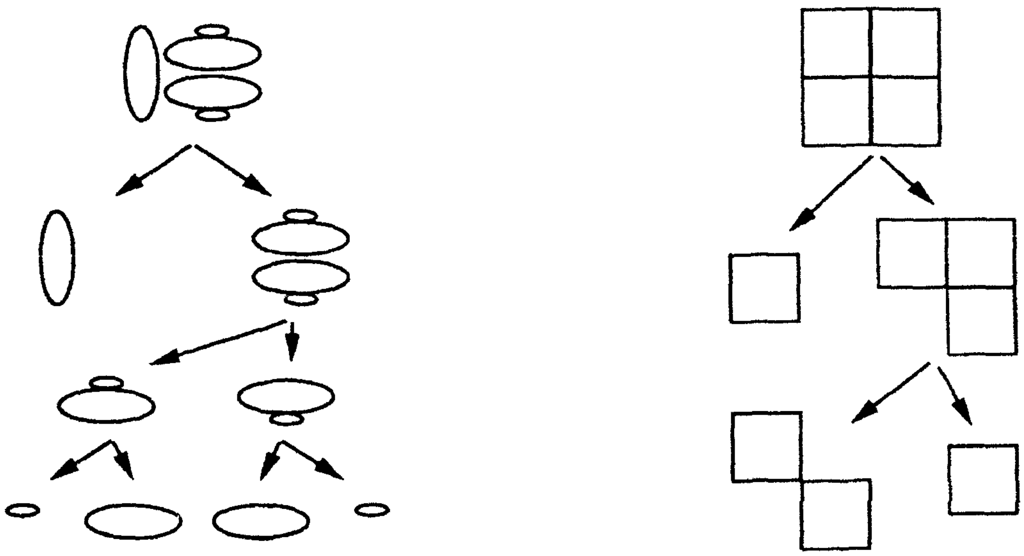

Figure 3: Optimal solutions found by tree-structured backprop network.

task for both backprop and the tree algorithm described in this paper. The numbers were hand-labelled, and preprocessed into a 240 dimensional feature vectors. Various backprop architectures were trained using cross-validation and an independent testing set. The best backprop performance was 93.8% for a 240-10-10 network while the best tree performance was 95.4% for a tree using a 240-4-2 splitting network. Testing on this classification problem is continuing. More results will be presented at the NIPS-91 conference.

4 Summary and Conclusions

In this paper, we have seen that the CART tree growing methodology combined with a soft-competition splitting rule can generate robust tree-structure neural networks by identifying sub-tasks which have a minimum of *confusion* between them. Smart partition choices lead to less complex nets and to less of an impact from spurious minima. We have also seen preliminary results that the adaptive partitioning rule proposed in this paper can reduce the interference problem in a real-world classification task.

Acknowledgements

I would like to thank Leon N Cooper and Nathan Intrator for many helpful discussions, and Nestor Inc. for making the NIST data available to me.

This paper presents some results from ongoing research into hybrid network algorithms at the Brown University Center for Neural Science and Nestor Inc.

References

Breiman, L., Friedman, J., Olshen, R., and Stone, C. (1984). *Classification and Regression Trees.* Wadsworth and Brooks/Cole Adv. Books and Software, Pacific Grove, CA.

Hampshire, J. and Waibel, A. (1989). The meta-pi network: Building distributed knowledge representations for robust pattern recognition. Technical report, Carnegie-Mellon Univeristy.

Hinton, G. E. and Nowlan, S. J. (1990). The bootstrap widrow-hoff rule as a cluster-formation algorithm. *Neural Computation*, 2(3):355–362.

Intrator, N. (1991). Localized exploratory projection pursuit. In *Proc. Interface Symposium.*

Jacobs, R. A., Jordan, M. I., and Barto, A. G. (1990). Taks decomposition through competition in a modular connectionist architecture: The what and where vision tasks. *Cognitive Science.*

Jacobs, R. A., Jordan, M. I., Nowlan, S. J., and Hinton, G. E. (1991). Adaptive mixtures of local experts. *Neural Computation,* 3(2).

Perrone, M. P. (1991). A novel recursive partitioning criterion. In *Proceedings of the International Joint Conference on Neural Networks.* IEEE.

Press, W. H., Flannery, B. P., Teukolsky, S. A., and Vetterling, W. T. (1987). *Numerical Recipes: The Art of Scientific Computing.*

Reilly, R. L., Scofield, C. L., Elbaum, C., and Cooper, L. N. (1987). Learning system architectures composed of multiple learning modules. In *Proc. IEEE First Int. Conf. on Neural Networks.*

Sanger, T. D. (1991). A tree-structured algorithm for reducing computation in networks with separable basis functions. *Neural Computation*, 3(1):67–78.

Sankar, A. and Mammone, R. (1991). Neural tree networks. In Mammone, R. and Zeevi, Y., editors, *Neural Networks: Theory and Applications.* Academic Press.

Sontag, E. D. and Sussmann, H. J. (1988). Backpropagation can give rise to spurious local minima even for networks without hidden layers. Technical report, Rutgers Univeristy.